Statistical Learning: Assignment 1

```
library(knitr)
```

```
poly_regression <- function(n){
  #generate training data
  x <- runif(n, min=-3, max=3)
  y <- 8*sin(x) + rnorm(n)
  train <- data.frame(x, y)

  ## generate test data:
  test_size <- 10000
  xtest <- runif(test_size, min=-3, max =3)
  ytest <- 8*sin(xtest) + rnorm(test_size)
  test <- data.frame(x = xtest, y = ytest)

  degrees <- c(3, 15)
  train_err <- test_err<-rep(NA, length(degrees)) # NA NA
  train_pred <- matrix(NA, nrow = length(degrees), ncol = n)

  i <- 1

  for (d in degrees){
    fit <- lm(y ~ poly(x, degree=d), data = train)
    train_pred[i,] <- predict(fit, newdata = train)
    test_pred <- predict(fit, newdata = test)
    test_err[i] <- mean((test$y - test_pred)^2)
    i <- i +1
  }
  print(test_err)


}

result1 <- round(poly_regression(50), digits=2)
```

```
## [1] 1.367724 1.318609
```

```
result2 <- round(poly_regression(10000), digits=2)
```

```
## [1] 1.1738647 0.9913127
```

The two values of mean squared error (MSE) for a training set of size of 50 are 1.37 and 1.32 for degree 3 and degree 15 respectively. Additionally, the two values of mean squared error for a training set of size of 10000 are 1.17 and 0.99 respectively.

1. As you can see the MSE for training set of 10000 is smaller for a degree of 15 than a degree of 3. Therefore the predictive function f is a function of degree 15 and its MSE is 0.99.

2. As shown below the two values of MSE for size of 50 are 1.37 and 1.32 for degree 3 and degree 15 respectively. Additionally, the two values of mean squared error for a training set of size of 10000 are 1.17 and 0.99 respectively. We notice that there is a significant increase in MSE from degree 3 to degree 15 when n = 50 but a decrease in MSE from degree 3 to degree 15 when n = 10000. This is because with 15 parameters, th amount of existing data in case 1 (n=50) is insufficient and we are thus underfitting the data, causing our model to be too simple to find the pattern accurately and thus leading to a high bias. Comparatively in the case where n = 10000, a larger training set of 10000 allows a more accurate prediction of the model (the bias decreases and variance increases) and therefore the difference between the actual value of y and expected value of y (f(x)) decreases, allowing us to reach the sweet spot where the sum of bias and variance is minimum and MSE is minimum.

The polynomial regression function can be seen above.